Account Management on a Large-Scale HPC Resource

Brett Bode, Tim Bouvet, Jeremy
Enos, Sharif Islam
National Center for Supercomputing
Applications,
University of Illinois, Urbana, IL.
Email: (brett, tbouvet, jenos,
mislam)@illinois.edu

ABSTRACT

Blue Waters is the largest system that Cray has built and operates in a very open network environment. This paper will discuss the design of the Blue Waters logical administrative network and how that design provides a secure and reliable environment that separates the user and administrative access paths. The paper will then describe how accounts and other user and project information is provisioned efficiently across its 27,000+ nodes.

CCS Concepts

• Security and Privacy→Security Services • Computer Systems Organization→Dependable and fault tolerant systems and networks.

Keywords

Blue Waters; Secure administrative networks; Account provisioning.

1. INTRODUCTION

The Blue Waters system [2-3] shown in Figure 1 is the National Science Foundation's track 1 system targeting high-end scientific computing projects. Blue Waters provides high-end computing to a wide variety of open, peer-reviewed science, engineering and education projects. The system consists of a 27,648 node Cray XE/XK computer utilizing Cray's Gemini based 3D torus high-speed interconnect. In addition, 26PB of storage is provided by a 432 node Cray Sonexion Lustre appliance and an ~80 node set of external servers provides login, data transfer and tape storage. Finally, the Cray XE/XK system is mirrored in one-rack, 96 node test system with its own control workstation.

The mission of Blue Waters is to provide high-end computational science support to a very wide range of scientific disciplines. The majority of the time on Blue Waters is allocated by NSF via a normal peer-reviewed proposal process. While many science teams are familiar to supercomputing centers around the country others are in emerging fields that are new to the computational science community. Science team members can be located virtually anywhere. Some teams utilize very traditional workflows contained entirely within the Blue Waters system while other teams have complex workflows that require connectivity from the compute engine to systems external to Blue Waters.

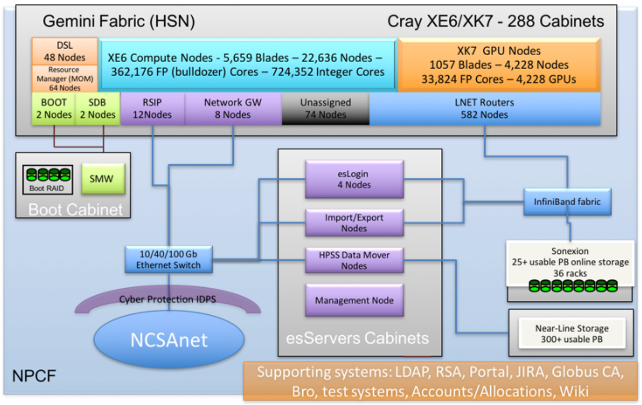

Figure 1: The Blue Waters System

Large computational science problems often require very large input and output data sets. Some of those data sets, potentially PBs in size must be transferred to/from Blue Waters. Blue Waters supports those large data flow requirements through a set of 25 servers dedicated to data transfer to/from the Lustre file systems and additional 50 servers acting as data movers for the HPSS environment both connected to nearly 400 Gb/s of WAN bandwidth to national and international backbone networks in Chicago. It is not practical to operate traditional firewalls or other packet inspection/filtering technologies at 100 Gbps link speeds so instead NCSA continues its practice of operating a very open network with many systems open to the outside network protected by a large Bro [4] based instruction detection and mitigation system. During an average month it is common for the system to block (black hole route) in excess of 250,000 external hosts.

Security is an issue with any network attached system, but even more important for a high-profile system such as Blue Waters. A key security measure included in the overall security profile is the use of RSA One-Time-Password (OTP) fobs for authentication. While no one likes OTP it largely eliminates the single largest source of compromises on previous systems, stolen credentials. While NCSA had utilized two-factor login for administrative access previously, Blue Waters is the first system at NCSA to utilize two-factor devices for regular user accounts requiring significant additional process development. In addition to monitoring all external traffic the NCSA security system also collects logs for many hosts and includes keystroke logging for the login hosts.

2. NETWORK DESIGN

The Blue Waters logical network layout shown in Figure 2 was designed taking into account many key factors. First, NCSA operates a very open network environment with no network firewalls in the network path to the general user accessible nodes (logins and data transfer nodes). Second, with the test system, external nodes, storage servers and main system there are no less than four separate administrative domains with the external servers and storage having additional redundant administrative servers. In addition, the Sonexion storage servers are not intended to allow site customization and are not properly configured (from a security point of view) to be exposed to a user accessible network. Third, a good security practice is to prevent a privilege escalation on a user accessible node from spreading to other nodes.

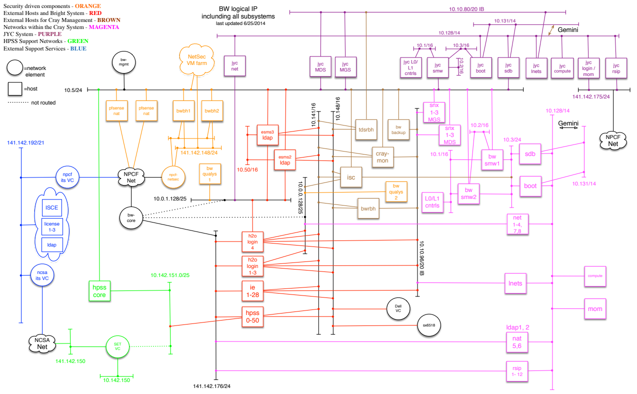

Figure 2: Blue Waters Logical Network Diagram

2.1 Bastion Hosts

A good practice in any environment, but an absolute necessity given the complexity of the various Blue Waters administrative servers is the use of bastion hosts[1]. The Blue Waters bastion hosts are two simple Linux servers that are setup to be independent of all other Blue Waters infrastructure that are connected to the general access NCSA network and the Blue Waters private administrative network. While in some setups this sort of host is a logical bounce host, meaning the backend network is not physically separate, in the Blue Waters case the bastion hosts are the only bridge point to the public network allowing outside logins. A true bastion host provides an additional layer of security since the administrative networks including the actual administrative servers can be isolated from the public network. It is also critical to keep the bastion host very simple providing only basic communications to limit possible vulnerabilities. Accounts on the bastion hosts are manually provisioned to the bastion after review of the need for administrative access and are automatically flagged for removal upon account closure. Two separate hosts are provisioned to ensure continuous availability while allowing the bastions themselves to be updated. The bastions can provide a modest staging area for transfers into the administrative network, but separate firewall systems also allow outbound traffic to selected external systems. This access allows for outbound email and for the normal Linux system update tools to operate without requiring extra administrative effort.

2.2 Privilege Escalation

A good security practice is to limit the spread of a compromise by limiting the ability of root on one system from getting root on its neighbor. In Blue Waters that is accomplished by requiring all administrative access to start on an administrative machine that can be reached only via the bastion hosts. Administrations utilize their normal user account to login to the bastion (using the previously mentioned two-factor login) and then login to the administrative host of choice depending on the required task. The administrator can sudo to root on the administrative host and then can ssh out to the managed nodes as needed. Root access is only allowed from the administrative host out to the edge systems. The reverse path is not allowed, nor is lateral access between edge systems.

2.3 Privilege Isolation

Even with the previously discussed measures each administrative domain is kept separate with access branching out from the bastion hosts and no access between the branches. Thus, it is not possible for a privileged account on the test system to administratively connect to administrative hosts on the main system and vice versa. As will be discussed in the next section the same account management is used across all systems including the administrative hosts with access controlled by groups (and by the requirement for a bastion host account). The access control groups allow separate access lists for each administrative domain giving us the flexibility, for example, to allow administrative access to the test system without providing it to the production system. The only significant exception to the mixing of administrative networks is a host called the Integrated System Console (ISC) that is used to centralize the logging and metric collection from throughout the Blue Waters super system[7].

3. ACCOUNT AND PROJECT MANAGEMENT

Managing accounts and projects on Blue Waters is similar to other HPC systems except that the accounts need to be efficiently distributed to 27,000+ nodes. In addition to regular user information, groups are used to group users into allocated projects. An external portal allows the project lead for each project to add/remove users. A custom database contains information about all current and past users and projects including all usage information. The external portal drives changes to that database and then need to automatically flow out to the full system. The project groups are also used to grant access to the major components of the system. That access control is done via the standard /etc/security/access.conf file. Projects are added and removed from access lines in access.conf as they are awarded time on the system or expire. An example of the access.conf file is below. The current actual file has over 200 groups listed.

```
+ : TRAIN_aaaa TRAIN_bbbb : 141.142.xxx.xx/32
- : ALL EXCEPT root crayadm globus bw_staff PRAC_cccc
ILL_dddd … : ALL
```

Figure 3: Sample access.conf File Syntax

However, distributing the access.conf file to nearly 80 hosts (logins and data transfer systems) is also impractical without automation. The solution was to make use of the previously mentioned ISC host. Changes are made on the ISC host via a web interface causing master access.conf, motd and an ssh banner file to be generated. Each host supporting external logins pulls those files from ISC once per minute. This interface is used not only for routine changes, but since it provides a very fast turnaround it is also used to update access control during system outages or before and after system maintenance.

Most groups are automatically generated and updated based on projects. However, there are also a small group of projects that are manually updated and are used for administrative access within the previously described administrative network. Those groups allow provided administrative privileges on each administrative domain.

3.1 LDAP

Prior to Blue Waters, NCSA used the state of the art (for 1990) method of account distribution of pushing out a password file from a central host via rsync or other sync method. Given the previous discussion of the complexity in the system administrative domains clearly makes the synced passwd file method unwieldy. Using the method described previously for access.conf would also be unlikely to scale to thousands of nodes. Instead based in part on two staff members previous experience LDAP was chosen to manage user and group information. LDAP has been proven at very large scale in traditional IT, but HPC resources can have much larger synchronized bursts of client requests (for instance at the time of a large job startup). LDAP allows user and group information to be provided uniformly across all parts of the super system despite different OSs and security levels including the Lustre file system servers that allow very limited on server changes. Changes are propagated very quickly limited only by the client cache time out. The previously mentioned custom database drives changes through custom scripts to make changes to LDAP. If an organization already had an existing LDAP system that could be used instead.

Standard LDAP fields are used in both normal and custom ways. For example, the description field for groups contains the short project description used for all Blue Waters communications. Email addresses for each account are populated and are changeable by end users via the user portal. Having email addresses readily available also allows script driven emails for a variety of notifications including producing emails to all "active" system users.

3.1.1 LDAP Scalability and Reliability

A single LDAP server can handle quite a bit of traffic. However, to scale out to large client count and to introduce some redundancy for fault tolerance LDAP slave servers should be used. In the Blue Waters LDAP setup, a VM outside of the Blue Waters network is used as the single LDAP master. All changes to LDAP are made on this host, but it is not configured as an LDAP server on any clients. The LDAP clients are configured to use a set of seven LDAP slave servers depending on their network location. The slave servers are in pairs, two on VMs external to the system, two on servers inside the Blue Waters administrative network, two on service nodes in the XE/XK compute fabric and one in the test system compute fabric. Clients are configured to connect to a nearby slave server and a second server as a backup.

Replication is accomplished using *syncrepl*. This means that replication is initiated by the client through the ldaps protocol on port 636 (Pull), and no special daemon and very little configuration on the master is required. Every object in the LDAP database has a timestamp associated with it (via the createTimestamp and modifyTimestamp attributes), so the replica-server just queries the master-server for changes since the newest record in the replica's database and updates changes every five minutes. If the replica database gets corrupted it is simple to recreate the database directory and copy over DB_CONFIG file from the corrupt directory. Restart LDAP on the slave and it will download the entire database from the master-server. The replicas are strategically placed on internal networks in close proximity to the compute and service nodes and includes a failover pair. There are also off cluster replicas in close proximity to the external mover nodes and Sonexion Lustre servers. In this configuration the system can withstand a master server failure and or network failures while maintaining full production. It also allows for rolling upgrades without downtime.

3.1.2 Extending LDAP

One advantage of running an independent LDAP infrastructure is that we are free to extend the standard LDAP schema as needed, though it is also possible to extend LDAP when tying into a central infrastructure. We have chosen to extend LDAP to track additional project information, both user and project quotas and the gridmap entry for the user. A field for a "picode" is used to indicate the PI or project leader, useful for correspondence and for staff to associate project members. The modifications to store quotas added a total of 12 items across the user and group definitions to store hard and soft block and inode quotas for each of the three BWs file systems. An example of these extensions are shown below.

```
dn: cn={5}inetorgperson,cn=schema,cn=config
changetype: modify
add: olcAttributeTypes
olcAttributeTypes: {11} ( 1.3.6.1.4.1.4203.666.1.44 NAME
'homeisquota' DESC 'Home Quota Value' EQUALITY
integerMatch SYNTAX 1.3.6.1.4.1.1466.115.121.1.27 SINGLE-
VALUE )
```

Figure 4: Example LDAP Schema Modification

Initial account creation is done entirely in LDAP. When the user logins for the first time the home and scratch directory are created and user/group file system quotas are set via a customized PAM module. At each login the quotas are compared with LDAP and adjusted if a change has been made to LDAP. The gridmap entry is used by gsissh and gridFTP for access via certificates (limited to certificates signed by an authority requiring two factor authentication). Traditionally a flat file is generated that must be updated as users are added and removed. Our approach is to store that map entry in LDAP and have modified gsissh and gridFTP clients to pull the entry from LDAP. The functionality to retrieve those entries from LDAP requires installation of Gridmap_Callouts[6].

3.2 Account and Project Removal

The majority of Blue Waters projects last one year. After a project expires there is an additional 90-day grace period to allow data access and transfer off of the system. Accounts and Projects are removed from the system for two primary reasons, security and storage. A good security practice is to remove access to inactive accounts and since users and projects on Blue Waters have generous storage quotas, that storage needs to be reclaimed for active projects. Inactive accounts can be an even larger security problem on systems using traditional passwords due to poor password protection practices of users. The use of two-factor authentication partially mitigates that problem, but there is an annual cost to maintain each token. Accounts can be removed from the system individually or as part of a full project removal. In addition, accounts can be removed from a single project, but remain on the system or be removed entirely. Individual account removals are initiated by the project lead via the user portal. When a removal is requested the system automatically removes the account from the requested project. If the account has no other active account, full account removal follows. That process is partly automated, but still involves manual steps to remove the user's storage and their two-factor access. Project removal is similar, but is only done after multiple notifications and has the additional step of removing project data. Currently when an account is removed it is fully deleted from LDAP. An alternative strategy could include moving the account to a "deactivated" LDAP space preventing access, but continuing to allow UID mapping to work.

3.3 Training Accounts

A key account management challenge for centers is dealing with potentially high account volume, short-duration projects used for training or education. This problem is compounded by the use of RSA one-time password tokens which have a high cost in terms of both the staff time to assign and distribute as well in the cost of the license and fobs. Since NCSA participates in workshops with potentially hundreds of participants a scalable alternative was needed.

The solution in use today was a limited bypass of the OTP requirement. When a training project is provisioned generic (instr001, tra0023, etc) accounts are provisioned into LDAP and a regular password is generated for each account for a VM specialized for accessing Blue Waters. These accounts are only allowed to connect to Blue Waters through that VM and during the period of the workshop or class. They authenticate with GSISSH certificates created for the exact period of the class using an offline CA specific for this purpose. Upon login, the user is presented with links to the Terms of Service that they must agree to before going further. The instructor is responsible for communicating appropriate use and the scope of the work, including any information about export control. The instructor will also securely distribute account information with unique passwords to the students, provided to the instructor. The following is an outline of the workflow.

1. Create an LDAP group and project for the specific class, and generate certificates for the lifetime of the class.
2. Enable accounts on the VM with gsissh and the pre-generated certificates installed for each account.
    1. This VM will have the motd banner about the ToS and the click-through on the first time.
    2. Login nodes will be configured to only accept connections for those accounts from that VM (use *access.conf* to limit logins by incoming IP)
3. Generate random local passwords for all the education accounts on the VM and distribute via encrypted pdf to the instructor and decrypt password through alternate delivery method to ensure security.
4. On the first day of instruction, enable the LDAP group for access.
5. After the final day of instruction, disable the LDAP group and accounts. Certificates will expire on their own.
6. Purge home and scratch directories on lustre for each account after class ends

3.3.1 Education Allocation Certificate Generation

For each class a new set of certificates, signed by the offline NCSA CA_BW certificate authority, needs to be generated and deployed to the secure VM server. Most of the certificate creation procedures are done on a shared drive /dev/shm of an internal server that has large memory available and thereby won't swap memory to disk in any normal scenario. Solely using the memory prevents certain sensitive information, for example the offline CA's private key, from ever being written to a hard drive in an unencrypted format. OpenSSL is used to generate and sign the certificates for each of the accounts required for a given event. The start and end times must be less than ninety days and scoped to the actual training timeframe. Scripts are used to generate the certificates in bulk and create a tarball that is transferred to the secure VM. The shred command is used to safely remove files and from /dev/shm following successful generation and transfer of the certificates to the VM machine. The certificates are deployed under local training accounts on the secure VM that will be used for the training event.

Use openssl to check the certificate validity of deployed certificate "openssl x509 -in ~account001/.globus/usercert.pem -text" (verify dates on certificate)

3.3.2 Creating Passwords and Encrypted PDF Account-Sheet Files

The process to assign education user account passwords on the secure VM and record them into printable pages is fairly automated. The random passwords are generated for the training group members and their account enabled in the /etc/shadow file. An encrypted pdf is generated for each site location containing unique password sheets. The pdfs are emailed to each site moderator to be distributed during the training session. The pdf decryption code is relayed to each moderator through text message or phone call.

3.3.3 Blue Waters Login Node Access

Blue Waters has three login nodes that are in a round robin ip configuration. Students login with username and password to the secure VM. Upon login, they will be presented with links to the Terms of Service that they must agree to before going further. The training accounts are configured with rbash as a default shell on the VM system. The /etc/profile ends with a line calling /usr/sbin/bounce. The bounce script executes "ssh -Y bw" for the instructor and training accounts. The login nodes have gsi-opensshd [5] configured to accept specific GSISSH certificates and RSA two factor authentication. The /etc/security/access.conf is configured to allow the LDAP training project scoped to the ip of the secure VM.

```
#!/bin/bash

if [[ $USER =~ tra[0-9]+ ]]; then

ssh -Y bw

kill `cat /proc/$PPID/status|grep PPid|awk '{print $2}'`

fi

if [[ $USER =~ instr[0-9]+ ]]; then

ssh -Y bw

kill `cat /proc/$PPID/status|grep PPid|awk '{print $2}'`

fi
```

Figure 5 Secure VM Bounce Script

The students access the secure VM which in turn bounces them onto one of three identical login nodes that have shared lustre file systems.

4. CONCLUSIONS

Managing account and project information in an efficient and secure fashion is a challenge for all computational systems. This paper has presented the Blue Waters approach to solving these challenges through the use of security best practices in setting up an administrative network and in the use of LDAP. While other systems are likely smaller in scale the same techniques can be applied effectively on virtually any system.

5. ACKNOWLEDGMENTS

This research is part of the Blue Waters sustained-petascale computing project, which is supported by the National Science Foundation (awards OCI-0725070 and ACI-1238993) and the state of Illinois. Blue Waters is a joint effort of the University of Illinois at Urbana-Champaign and its National Center for Supercomputing Applications. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

6. REFERENCES

[1] Bastion Host Definition https://en.wikipedia.org/wiki/Bastion_host/. Accessed: 2016-08-24.

[2] Blue Waters: Sustained Petascale Computing https://bluewaters.ncsa.illinois.edu/. Accessed: 2016-08-22.

[3] Brett Bode, Michelle Butler, Thom Dunning, William Gropp, Torsten Hoefler, Wen-mei Hwu, and William Kramer. The Blue Waters Super-System for Super-Science. Contemporary HPC Architectures, Jeffery Vetter editor. Sitka Publications, November 2012. Edited by Jeffrey S . Vetter, Chapman and Hall/CRC 2013, Print ISBN: 978-1-4665-6834-1, eBook ISBN: 978-1-4665-6835-8

[4] Bro: The Bro Network Security Monitor https://www.bro.org/. Accessed: 2016-08-23.

[5] Globus Toolkit: GSI-OpenSSH http://toolkit.globus.org/toolkit/docs/5.0/5.0.4/security/openssh/. Accessed: 2016-08-23.

[6] GridMap Callouts: Support for placing GridMap contents in LDAP https://github.com/JasonAlt/Gridmap_Callouts/. Accessed: 2016-08-24.

[7] Semeraro B. D., Sisneros R., Fullop J., and Bauer G.H., "It Takes a Village: Monitoring the Blue Waters Supercomputer", IEEE Cluster Workshop on Monitoring and Analysis for High Performance Computing Systems Plus Applications 2014.